

REMARKS

Claims 1-21 are pending in the present application. Claims 1-21 were canceled; and claims 22-39 were added. Reconsideration of the claims is respectfully requested.

Amendments to the specification were made to correct errors of a typographical nature. No new matter has been introduced by the amendments to the specification.

I. 35 U.S.C. § 112, First Paragraph

The office action has rejected claims 1, 6, 8, 13, 15, and 20 under 35 U.S.C. § 112, first paragraph, as containing subject matter which was not described in the specification in such a way as to enable one skilled in the art to which it pertains to make and/or use the invention. Claims 1-21 have been cancelled, and the rejection of claims 1, 6, 8, 13, 15, and 20 under 35 U.S.C. § 112, first paragraph, is now moot.

II. 35 U.S.C. § 112, Second Paragraph

The office action has rejected claims 1, 6, 8, 13, 15 and 20 under 35 U.S.C. § 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicants regard as the invention. Claims 1-21 have been cancelled, and the rejection of claims 1, 6, 8, 13, 15 and 20 under 35 U.S.C. § 112, second paragraph is now moot.

III. Double Patenting

The office action has rejected the claims 1-21 of the subject application under nonstatutory double patenting in view of U.S. Patent No. 6,546,365 to Gajda et al. (hereinafter Gajda). Claims 1-21 have been cancelled and claims 22-39 have been added. Newly added claims 22-39 are directed to a mechanism of managing messages of a distributed application. Particularly, claims 22-39 are directed to a method, computer program product, and a network of data processing systems in which a task issued by a client having a language of a first locale is parceled into subtasks that are distributed by a server node to other nodes in the network for processing. A task sent by the client to the server node includes an identifier of the language of the first locale. When a node completes a subtask, a completion result of the sub-task is translated by the node that

performs the sub-task based on the identifier of the client (first) locale if the node is capable of translating the result into the language of the client locale. The translated result is then returned to the server node that conveyed the sub-task to the node that generated the completion result and is consolidated with other results corresponding to other completed sub-tasks. In the event that the node that completed the sub-task is not adapted to translate the result into the language of the client locale, the result is passed to the server node or, additionally, one or more intervening nodes between the node that processed the sub-task and the server node that distributed the sub-task. Accordingly, a result message formed from consolidating sub-task results may include translated results of sub-tasks as well as non-translated results of sub-tasks (See Page 3, Lines 8-23; Page 10, Lines 4-10, and Lines 16-20; Page 11, Lines 12-22, and Line 25; Page 12, Line 5; Page 13, Lines 3-7, Lines 9-13, and Lines 23-28). No new matter has been introduced by the addition of claims 22-39.

Gajda is directed to a mechanism to construct programs for use in multiple-language environments. A message ID code is passed to a subroutine with a default string and a language. A translation file is opened that corresponds to the language passed to the subroutine and accesses a message that corresponds to the message ID. The message that corresponds to the message ID is then displayed.

Gajda is silent with regard to managing messages in a distributed application and is thus wholly insufficient to anticipate newly added claims 22-39. Withdrawal of the double patenting rejections is respectfully requested.

IV. 35 U.S.C. § 102, Anticipation

The examiner has rejected claims 1-21 under 35 U.S.C. § 102 as being anticipated by U.S. Patent No. 5,724,503 to Kleinman et al. (hereinafter referred to as Kleinman). Claims 1-21 have been cancelled, and the rejection thereof is thus moot.

Claims 22-39 have been added and are directed to a method, computer program product, and network for managing results of a distributed application in a locale independent manner. For example, newly added claim 22 recites the following:

22. (New) A method for managing results in a locale independent manner in a multi-node networked data processing system, the method comprising:

receiving, at a first node, a task to be performed sent from a client node connected with the first node, wherein the client node has a language of a first locale associated therewith and the task includes an identifier of the language of the first locale;

responsive to receiving the task, parceling, by the first node, the task into a plurality of sub-tasks that each may be independently executed, wherein the plurality of sub-tasks include a first sub-task;

conveying the first sub-task having the identifier of the first locale to a second node connected with the first node for processing of the first sub-task, wherein the second node has a language of a second locale associated therewith that is different than the language of the first locale;

responsive to receiving by the first node a first message resulting from completion of the first sub-task, translating, by the first node, the first message into the language of the first locale;

consolidating, by the first node, the first message translated into the language of the first locale and at least one message resulting from completion of a second sub-task processed by a third node into a results message; and

sending the results message to the client node.

As described in the subject application, a client sends a task to be performed to a server node connected with the client (Page 10, Lines 16-17). The task includes an identifier that indicates a language of a locale in which the client is located. The server node receives the task and parcels the task into sub-tasks that are each to be independently executed (Page 10, Lines 18-20 and Lines 28-31). One or more of the sub-tasks are then conveyed to other nodes in the network for processing. Upon completion of each sub-task, the node processing the sub-task generates a result, for example a result array (Page 11, Lines 1-3). A message of the result of a sub-task may then be translated from a language of a locale of the node that processed the sub-task to the language of the locale of the client based on the identifier that is retrieved by the node that processed the sub-task (Page 15, Lines 1-13). If the node that processed the sub-task is not adapted to produce or translate a result message in the language of the locale of the client node, the untranslated message is returned to the next node in the network, i.e., the next upstream node in the network (Page 15, Line 19-23). If an intermediate node exists between the processing node and the client node, a result message in a language other than the language of the client locale may be translated by the intermediate node if the intermediate node is suitably adapted for such translation. If not, the message is then forwarded toward the client node that distributed the sub-tasks, for example added to a

result message of the intermediate node and then sent on to the next intermediate node (if one exists) or to the client node. Accordingly, a result message that is produced in a language other than the language of the locale of the client node may be translated by the node that produced the result message or any node in between the node that produced the result message and the client node, for example the server node that distributed the sub-tasks (Page 11, Lines 12-15). For example, Figure 4 shows the following:

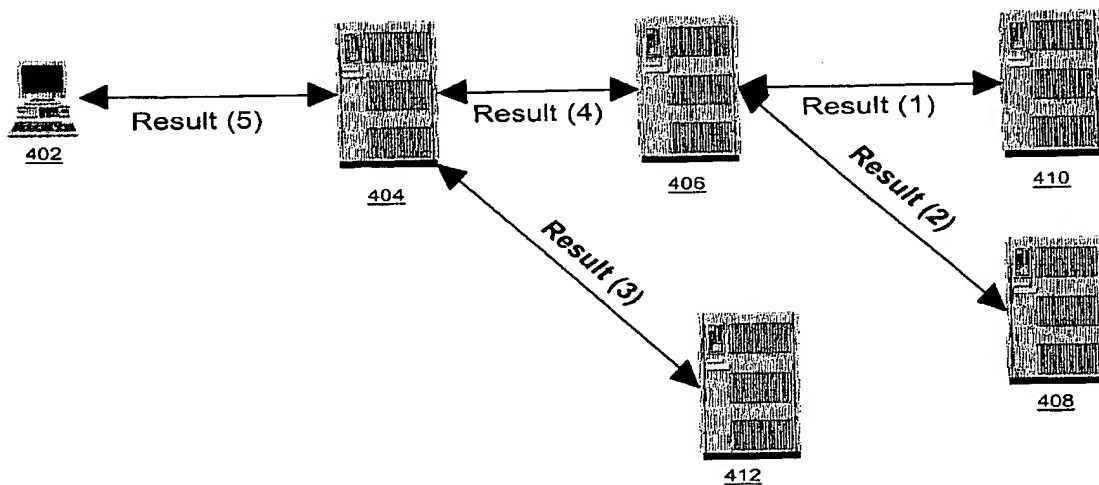


Figure 4
AUS9-2000-0458-US 1

As can be seen, a client node (402) within a distributed network connects and sends a task to be completed to a server node (404). The server node then parcels the task into sub-tasks that are distributed to other nodes (406-412) within the network. An identifier of a language of the client node is included in the task and sub-tasks. As the sub-tasks are completed by the nodes receiving a sub-task, respective results including messages are generated that are to be returned to the server node that distributed the sub-tasks. As described in the application, if a node that completes a sub-task produces message text that is in a language different than the locale of the client node and if the node is adapted to translate the message text into the language of the locale of the client node, the translation is performed at the node that completes the sub-task. The result with message text produced from processing of a sub-task is conveyed to the server node that

distributed the sub-tasks. If a result returned to the server node that distributed the sub-task is in a language of a locale different than the language of the locale of the client node (that is, if the node that produced the sub-task and any intervening nodes was unable to translate the result), the server node (if suitably adapted) will translate the result into the language of the locale of the client node. If the server node is not adapted to translate the message text of the result into the language of the locale of the client node, the message text of the result is consolidated with other sub-task results and returned to the client node in the same language as received by the server node from the processing node.

With regard to U.S. Patent No. 5,724,503, Kleinman shows a computer system and method used for converting an exception identifier received by a host computing system from a remote device to a message string that corresponds to the exception identifier. For example, Kleinman states the following:

...the present invention contemplates *storing the actual exception text*, in the language of the client, *on the client machine*. This requires the server to send a unique exception identifier (unique across all possible applications running on any node supported by the object request broker). When a server returns an exception identifier, the client first uses the exception identifier to derive an exception tag. The exception tag can then be used to locate the host exception file *on the client* computer that provides a base language (e.g. English) explanation. (*emphasis added*)

Kleinman, Column 6, Lines 54-63

Thus, Kleinman provides a mechanism for an exception explanation by storing exception text at the client and locating the exception text based on an exception identifier. For example, Figure 5 of Kleinman shows the following:

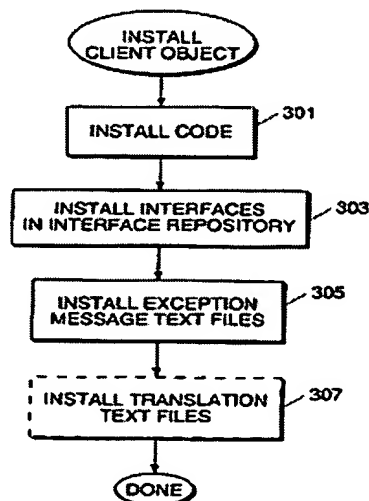


Fig. 5

As can be seen, Kleinman requires installation of translation text files (step 307) on the host for performing exception translation when the native language of the host machine's user is not the base language of the exception file (Column 12, Lines 9-13).

Additionally, Kleinman describes a mechanism for converting an exception string in a base language that is not a preferred language of a client. For example, Figure 3 of Kleinman shows the following:

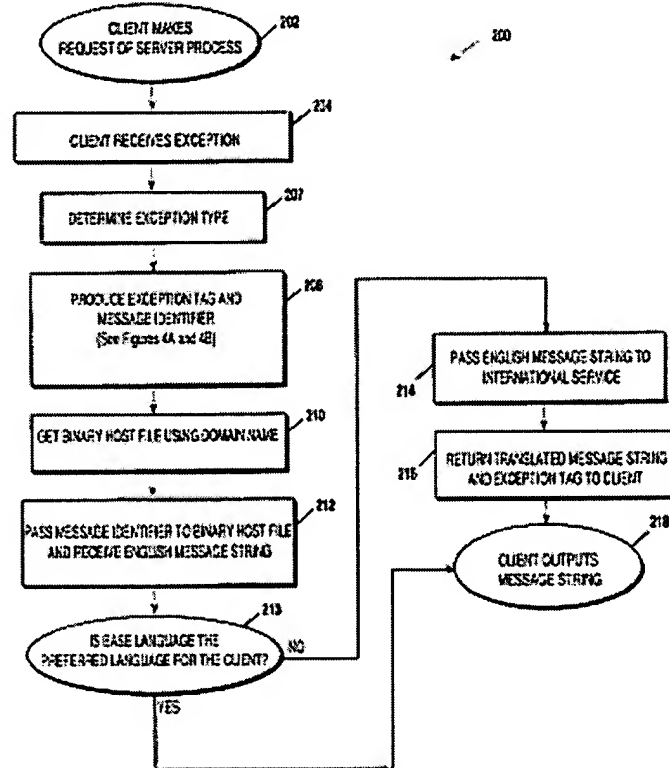


Fig. 3

As can be seen, an explanation message string of an exception is received from an exception message file (step 212), and an evaluation is then made to determine if the message string is in the preferred language of the client (step 213). If the message string is not in the preferred language of the client, the client passes the message string to an international service function for translation of the message string (step 214).

Thus, Kleinman requires the host to first receive a message string and evaluate whether the message string requires translation. The client then sends the message string to a translation service if the message string is not in a preferred language.

Notably, Kleinman is silent with regard to managing messages of a distributed application and does not facilitate translation of a message string or other text, such as a task result response, prior to delivery of the message string to the host.

The present invention provides the advantage of performing any available translations during return of a result message to the client thereby eliminating any need for the client to interact with a translation service after receiving the message string. Notably, the mechanism described and claimed in the subject application provides for the client to include an identifier of a language of the locale of the client. Additionally, a node that performs processing of a sub-task parceled from a client task will translate a result of the completed sub-task if the node is suitably adapted. If the node is not adapted to translate the sub-task completion result into the language of the client locale, the result is passed to the next (if any) intermediate node until the result is received by the server node that parceled the task. If any intermediate node or the server node that distributed the sub-tasks is adapted to translate the sub-task completion result to the language of the client locale, the translation is performed upon receipt of the sub-task completion result by the suitably adapted intermediate node or server node. By providing the locale identifier of the client to each node involved in processing of sub-tasks parceled from the client task, any node involved in returning a sub-task result to the server node that distributed the sub-tasks may translate a result received that has not yet been translated.

Thus, Kleinman fails to at least describe a mechanism for receiving, from a client node, a task to be performed at a node that parcels "the task into a plurality of sub-tasks," and for including "an identifier of the language of the first locale" of the client in the task. As Kleinman is silent with regard to distributed processing of an application task, Kleinman consequently provides no description or suggestion for conveying a "first sub-task having the identifier of the first locale" to a second node for processing of the sub-task. Moreover, as noted above, Kleinman requires the host or client to first receive a message string and evaluate if the message string is in the desired language - the host then communicates the message string to a translation service if the message string is not

in the desired language. Thus, Kleinman fails to describe or suggest a mechanism for "receiving a first message resulting from completion of the first sub-task" by the node that parceled and distributed sub-tasks from the task wherein the node that distributed the sub-task translates the first message into the language of the locale of the client and then sends the translated result to the client. Furthermore, as Kleinman is wholly silent with regard to distributed processing of an application task, no description or suggest is provided by Kleinman for consolidating "the first message translated into the language" of the locale of the client with "at least one message resulting from completion of a second sub-task" parceled from the task that is then sent to the client. Thus, Kleinman is wholly insufficient to anticipate or suggest the present invention as claimed by newly added independent claim 22.

Newly added independent claims 28 and 34 recite similar features as claim 1. Therefore, the same distinctions between Kleinman and the claimed invention in claim 22 apply for these claims. For the reasons described above, Kleinman does not contain all elements of independent claims 22, 28 and 34. Hence, Kleinman fails to anticipate the present invention as recited in claims 22, 28, and 34. Since claims 23-27 depend from claim 22, claims 29-33 depend from claim 28, and claims 35-39 depend from claim 34, the same distinctions between Kleinman and the claimed invention in independent claims 22, 28 and 34 apply for these claims. Additionally, claims 23-27, 29-33, and 35-39 claim other additional combinations of features not suggested by Kleinman. Consequently, it is respectfully urged that claims 22-39 are patentable over Kleinman.

Therefore, Applicants submit that claims 22-39 are patentable over Kleinman and such a notice is respectfully requested.

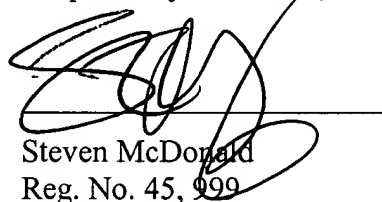
V. **Conclusion**

It is respectfully urged that the subject application is patentable over Kleinman and Gajda and is now in condition for allowance.

The examiner is invited to call the undersigned at the below-listed telephone number if in the opinion of the examiner such a telephone conference would expedite or aid the prosecution and examination of this application.

DATE: September 13, 2004

Respectfully submitted,

A handwritten signature in black ink, appearing to read 'Steven McDonald', is written over a horizontal line.

Steven McDonald
Reg. No. 45, 999
Yee & Associates, P.C.
P.O. Box 802333
Dallas, TX 75380
(972) 367-2001
Agent for Applicant